

09/727,191

MS150809.01/MSFTP177US

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1. (Currently Amended) A system to facilitate a remote user accessing an application across a stateless protocol, comprising:
a component for caching data associated with the remote user access, the data comprising state and/or user specific information; and
a component for managing memory storing the state and/or user specific information, wherein the user specific information and the state information are stored in a user context object; and
the memory managing component reclaims resources allocated to the user context object upon a determination that the user context object has not been accessed within a pre-determined threshold of time.
2. (Cancelled)
3. (Original) The system of claim 1, the user specific information comprising at least one of:
a record of views accessible to the remote user, a record of views recently displayed to the remote user, a record of folders recently accessed by the remote user, a record of URLs for folders commonly accessed by the remote user, a record of messages and/or folders moved and/or copied by the remote user, and email addresses for the remote user.
4. (Original) The system of claim 1, the state information comprising at least one of:
a record of one or more entities involved in a process requiring multiple requests from the remote user *via* the stateless protocol.

09/727,191

MS150809.01/MSFTP177US

5. (Original) The system of claim 4, the entities comprising at least one of:
memory locations, folders, directories, messages, objects, processes, threads,
records, files and data.
6. (Original) The system of claim 1, wherein the stateless protocol is HTTP.
7. (Original) The system of claim 1, the application comprising at least one of:
email, chat sessions, database programs, video games, web-enabled applications and
search engines.
8. (Cancelled)
9. (Cancelled)
10. (Currently Amended) The system of claim [[9]] 1, wherein the user context object
is assigned a globally unique identifier.
11. (Currently Amended) The system of claim [[2]] 1, wherein the memory managing
component manages one or more user context objects.
12. (Original) The system of claim 11, wherein the memory managing component can
locate a user context object *via* an addressing algorithm, the algorithm employing the globally
unique identifier, a locale identifier, a mailbox identifier and a security identifier.
15. (Currently Amended) The system of claim 1[[2]], wherein the resources include at
least one of memory, data communications devices, processor time and network bandwidth.
14. (Currently Amended) The system of claim 1[[3]], wherein the pre-determined
threshold period of time is one hour.

09/727,191

MS150809.01/MSFTP177US

15. (Currently Amended) The system of claim 1[[3]], wherein the pre-determined threshold period of time can be dynamically changed based, at least in part, on feedback concerning the usage of one or more user context objects.

16. (Original) The system of claim 1, further comprising a monitoring component, operable to feedback information concerning usage of one or more user context objects.

17. (Original) The system of claim 16, wherein more resources are allocated to a user context object when the feedback information indicates that the user context object has been utilized more than a first pre-determined threshold level.

18. (Original) The system of claim 17, the resources comprising at least one of memory, processor time, communication devices and network bandwidth.

19. (Original) The system of claim 16, wherein resources are reclaimed from a user context objects when the feedback information indicates that the user context object has been utilized less than a second pre-determined threshold level.

20. (Original) The system of claim 19, the resources comprising at least one of memory, processor time, communication devices and network bandwidth.

21. (Original) The system of claim 16, wherein resources are shifted between one or more user context objects based, at least in part, on feedback information.

22. (Currently Amended) A method for facilitating remote access to an application, the access occurring across a stateless protocol, comprising:

allocating memory to cache state and/or user specific information associated with the remote access;

caching the state information associated with the remote access of the application in the allocated memory; ~~and~~

09/727,191

MS150809.01/MSFTP177US

caching the user specific information associated with the remote access of the application in the allocated memory, and

selectively reclaiming the memory allocated to cache the state and/or user specific information associated with the remote access, the reclaiming occurring upon the allocated memory not being accessed within a pre-determined period of time.

23. (Cancelled) The method of claim 22, further comprising selectively reclaiming the memory allocated to cache the state and/or user specific information associated with the remote access, the reclaiming occurring upon the allocated memory not being accessed within a pre-determined period of time.

24. (Original) The method of claim 23, the pre-determined period of time being one hour.

25. (Original) The method of claim 23, the pre-determined period of time being dynamically adjustable based, at least in part, on feedback concerning the usage of the allocated memory associated with the remote access.

26. (Original) The method of claim 22, further comprising assigning a globally unique identifier to the memory allocated to cache the state and/or user specific information.

27. (Original) The method of claim 26, further comprising locating the state and/or user specific information *via* an algorithm, the algorithm employing the globally unique identifier, a locale identifier, a mailbox identifier and a security identifier.

28. (Original) The method of claim 22, the state information comprising at least one of:

a record of one or more entities involved in a process requiring multiple requests from the remote user *via* the stateless protocol.

29. (Original) The method of claim 28, the entities comprising at least one of:

09/727,191

MS150809.01/MSFTP177US

memory locations, folders, directories, messages, objects, processes, threads, records, files and data.

30. (Original) The method of claim 22, wherein the stateless protocol is HTTP.
31. (Original) The method of claim 22, the application comprising at least one of: email, chat sessions, database programs, video games, web-enabled applications and search engines.
32. (Original) The method of claim 22, further comprising:
receiving feedback information concerning usage of one or more user context objects;
and
allocating more resources to a user context object, based at least in part, on the feedback information.
33. (Original) The method of claim 32, further comprising:
de-allocating resources from a user context object, based at least in part, on the feedback information.
34. (Original) The method of claim 32, further comprising:
shifting resources between one or more user context objects, based at least in part, on the feedback information.
35. (Currently Amended) The system of claim 1, further comprising a [[A]] data packet adapted to be transmitted between two or more computer processes, comprising,
information related to facilitating remote access to an application, the access occurring across a stateless protocol, the information comprising:
state and/or user specific information associated with the remote access of the application.
36. (Currently Amended) A computer readable medium storing computer executable components of a system to facilitate a remote user accessing an application across a stateless

09/727,191

MS150809.01/MSFTP177US

protocol, the components including a component for caching data associated with the remote user access, the data comprising state and/or user specific information, the components including a component for selectively reclaiming the memory allocated to cache the state and/or user specific information associated with the remote access, the reclaiming occurring upon the allocated memory not being accessed within a pre-determined period of time.

37. (Original) The computer readable medium of claim 36 further including a component for managing memory storing the state and/or user specific information.

38. (Currently Amended) A computer readable medium storing computer executable instructions operable to execute a method for facilitating remote access to an application, the access occurring across a stateless protocol, the method comprising:

allocating memory to cache state and/or user specific information associated with the remote access;

caching the state information associated with the remote access of the application in the allocated memory; ~~and~~

caching the user specific information associated with the remote access of the application in the allocated memory; and

selectively reclaiming the memory allocated to cache the state and/or user specific information associated with the remote access, the reclaiming occurring upon the allocated memory not being accessed within a pre-determined period of time.

39. (Cancelled)

40. (Original) The computer readable medium of claim 39, further including computer executable instructions operable to dynamically reallocate resources to and/or from one or more user context objects based, at least in part, on feedback information received from one or more monitoring components.

09/727,191

MS150809.01/MSFTP177US

41. (Original) A system for managing information in an application accessed *via* a stateless protocol, comprising:

means for allocating memory to store state and/or user specific information associated with a remote user access of an application, the application being accessed *via* a stateless protocol;

means for caching, in the allocated memory, state information associated with the remote access of the application;

means for caching, in the allocated memory, user specific information associated with a remote accessor of the application;

means for locating the state and/or user specific information cached in the allocated memory; and

means for reclaiming the allocated memory when the memory has not been accessed within a pre-determined period of time.